

Pairwise Preference Regression on Movie Recommendation System

Rudy Prabowo ^{#1}, Agung Toto Wibowo ^{#2}, Rita Rismala ^{#3}

*Fakultas Informatika, Universitas Telkom
Bandung, Indonesia*

¹ rudyprabowo@live.com

² agungtoto@telkomuniversity.ac.id

³ ritaris@telkomuniversity.ac.id

Abstract

Recommendation System is able to help users to choose items, including movies, that match their interests. One of the problems faced by recommendation system is cold-start problem. Cold start problem can be categorized into three types, they are: recommending existed item for new user, recommending new item for existed user, and recommending new item for new user. Pairwise preference regression is a method that directly deals with cold-start problem. This method can suggest a recommendation, not only for users who have no historical rating, but also for those who only have less demographic info. From the experiment result, the best score of Normalized Discounted Cumulative Gain (nDGC) from the system is 0.8484. The standard deviation of rating resulted by the recommendation system is 1.24, the average is 3.82. Consequently, the distribution of recommendation result is around rating 5 to 3. Those results mean that this recommendation system is good to solving cold-start problem in movie recommendation system.

Keywords: recommendation system, Pairwise Preference Regression, cold-start problem, movies

Abstrak

Sistem rekomendasi membantu pengguna untuk memilih buku, artikel, film, musik, barang elektronik, dan yang lainnya untuk menemukan barang atau informasi yang paling menarik dan bermanfaat bagi mereka. Namun salah satu masalah yang dihadapi oleh sistem rekomendasi adalah cold-start problem. Pada cold-start problem sistem rekomendasi sulit memberikan rekomendasi kepada pengguna karena beberapa faktor. Cold-start dapat di kategorikan dalam tiga tipe yaitu merekomendasikan item yang sudah ada untuk pengguna baru, merekomendasikan item baru untuk pengguna yang sudah ada, dan merekomendasikan item baru untuk pengguna baru. Pairwise preference regression merupakan salah satu metode yang langsung mengatasi masalah cold-start, metode ini dapat memberikan rekomendasi tidak hanya pada pengguna yang tidak memiliki historical rating tetapi juga hanya memiliki sedikit informasi demografi. Dari hasil pengujian didapatkan nilai Normalized Discounted Cumulative Gain (nDGC) terbaik dari sistem adalah 0.8484 dan hasil rekomendasi yang dihasilkan oleh system memiliki standar deviasi 1.24 dengan rata-rata 3.82. sehingga hasil rekomendasi yang dihasilkan memiliki persebaran rating antara 5 sampai 3. Dari hasil tersebut juga dapat dilihat bahwa sistem dapat memberikan hasil rekomendasi yang bagus untuk mengatasi masalah cold-start.

Kata Kunci: sistem rekomendasi, Pairwise Preference Regression, cold-start problem, film

I. INTRODUCTION

THE great number of information availability and accessibility is impacted on the problem of how to filter the relevant information for users. For that reason, recommendation system emerges as one of the solutions in handling that problem by providing suggestion to help user in making decision [1].

In many cases, users rely on recommendation from another user, in the form of product review, news from social media or electronic media, survey, etc., to choose or buy a product, including choose a movie to watch.

With the increasing number of available movies recently, causing the audiences need to be selective to choose movies that they would like to watch.

Content-based filtering and collaborative filtering are common approaches in recommendation system [2]. Content-based filtering makes a profile for a user based on the items that have been rated by that user. Using this approach means that the system can recommend new item that never been rated before based on the profile. Collaborative filtering system relates the user with users group that has the same interest. Consequently, the system can make recommendation without depending on the characteristics of the items that have been rated [2].

One of the problems faced in recommendation system is cold-start problem. This problem may appear in a new recommendation system. When users are still new and the items provided in the system are only a few, so it is hard for the system to give recommendations; this is what they call cold-start problem. Cold-start problem can be categorized into three types; 1) recommending available items for new users, 2) recommending new item for new users, 3) recommending new item for new users [3].

There are many algorithm that have been made to solve cold-start, such as in [4] that proposed hybrid method to recommend items based on voting from four algorithms, they are: user-user collaborative filtering, content-based, demographic-based, and collaboration via content; in [5] was proposed online perceptron algorithm with multiple kernel functions combination that combine collaborative and content-based filtering. However, those methods did not directly solve the cold-start problem itself, it was because the user targets have made several ratings to the recommended items.

Pairwise preference regression is one of the methods that directly deal with cold-start problem. Pairwise preference regression can recommend not only for users with no historical rating but also for those with few demographic information [3].

II. LITERATURE REVIEW

A. Recommendation System

Recommendation system uses opinion from users' community to help users to effectively identifying preferred items based on their interest, from so many items provided [6]. In general, the problem in recommendation system is how to decrease the prediction rating for item that user never seen before. The prediction is usually taken from ratings that are given by users to other item. After estimating rating for items which is never seen by users, then we can recommend item with the highest rating to users [1].

Based on how recommendation is made, recommendation system is categorized into three approaches, those are [1]:

- content-based recommendations: user will be given item recommendation based on similarity from previously rated item by those users.
- collaborative recommendations: users will be given item recommendation based on other users that have the same preference.
- hybrid approaches: this method combines between Content-based recommendations and Collaborative recommendations.

B. Pairwise Preference Regression

In [3] has been proposed Pairwise Preference Regression method that directly deal with cold-start problem in recommendation system. In this method, predictive model connecting pair of vector x_u and z_i to rate r_{ui} using outer products, where every pair will be represented as $x_u \otimes z_i$, vector from CD with component $\{x_{u,a} z_{i,b}\}$ where $z_{i,b}$ will be notated as attribute of b from z_i and $x_{u,a}$ will be notated as attribute of a from x_u . For

example, value of $x_u = \{1, 0, 1\}$ then value of $x_{u,1} = 1$, $x_{u,2} = 0$, and $x_{u,3} = 1$ and value of $z_i = \{1, 1\}$ then $z_{i,1} = 1$ and $z_{i,2} = 1$. \mathcal{O}

Parametric indicator as a function of bilinear function from x_u and z_i can be seen in (1), where C and D are dimensionality from users and each attribute content; a, b are attribute index. The weight of variable w_{ab} independent from users and attribute content, and it indicates equation from two factor $x_{u,a}$ and $z_{i,b}$ in interaction. The indicator can be rewritten as (2), where \mathbf{W} is matrix with member $\{w_{ab}\}$, where \mathbf{w} is notated as stack of column vector from \mathbf{W} , and $z_i \otimes x_u$ is notated as outer product from x_u and z_i , column vector with member $\{x_{u,a} z_{i,b}\}$.

$$s_{ui} = \sum_{a=1}^C \sum_{b=1}^D x_{u,a} z_{i,b} w_{ab} \quad (1)$$

$$s_{ui} = \mathbf{x}_u \mathbf{W} z_i^\top = \mathbf{w}^\top (z_i \otimes \mathbf{x}_u) \quad (2)$$

The optimization problem in Pairwise Preference Regression is shown in (3), where \mathcal{O}_u is the index set of all items that have rated by user x_u , $n_u = |\mathcal{O}_u|$ the number of ratings given by the user x_u , and s_{ui} is preference score as defined in (2). To get an optimal weight, it can be calculated with close form in equation (4) where w^* as dependent attribute, and I is a CD by CD identity matrix.

$$\arg \min_w \sum_u \left(\frac{1}{n_u} \sum_{i \in \mathcal{O}_u} \sum_{j \in \mathcal{O}_u} ((r_{ui} - r_{uj}) - (s_{ui} - s_{uj}))^2 \right) + \lambda \|w\|_2^2 \quad (3)$$

$$w^* = \left(A + \frac{\lambda}{2} I \right)^{-1} B \quad (4)$$

$$A = \sum_u \sum_{i \in \mathcal{O}_u} z_i (z_i - \tilde{z}_u)^\top \otimes x_u x_u^\top \quad (5)$$

$$B = \sum_u \sum_{i \in \mathcal{O}_u} r_{ui} (z_i - \tilde{z}_u) \otimes x_u \quad (6)$$

$$\tilde{z}_u = \frac{1}{n_u} \sum_{i \in \mathcal{O}_u} z_i \quad (7)$$

C. Normalized Discounted Cumulative Gain

In [7] has been proposed an evaluation metric called Normalized Discounted Cumulative Gain (nDCG). nDCG measuring performance from recommendation system based on relevancy level from recommendation entity. The range is from 0.0 to 1.0, where 1.0 is represented as an ideal rank from entity. Equation (8) is formula to calculate nDCG. k is defined as maximum total value from recommended entity, R_{ui} is the actual rating from user u in the i -th rank item, U_T is a set of users test data, and $IDCG_k^u$ is the best possibilities from DCG_k^u for user u .

$$nDCG_k = \frac{1}{|U_T|} \sum_{u \in U_T} \frac{DCG_k^u}{IDCG_k^u} \quad (8)$$

$$DCG_k^u = \sum_{i=1}^k \frac{2^{R_{ui}-1}}{\log_2(1+i)} \quad (9)$$

III. RESEARCH METHOD

A. Data Set

Data set that used in this system was movie rating data set from Movielens site, namely 100K data set, it has 100.000 rating with total of 943 users and 1682 items. In the experiment, data set were separated into four partitions [3]. The partitions were divided randomly into a half as a new user and the other half as existed

user. And for data item, they were divided randomly into a half as new item and the other half as existed item. Fig. 1 illustrates how data were distributed. Part I (including existed data user and existed data item) is used as modeling data and part II (including new user data and existed item), part III (including existed data user and new item), and part IV (new data user and new data item) used as testing data.

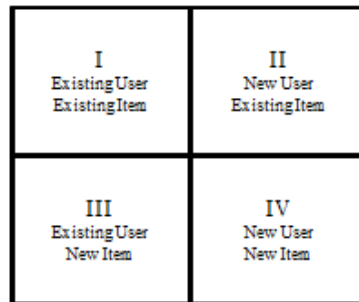


Fig. 1. Data partition [3].

B. System Design

The recommendation system was developed using Pairwise Preference Regression method. This system consists of two main processes that were modeling and testing. Fig. 2. shows illustration about general scheme of our system.

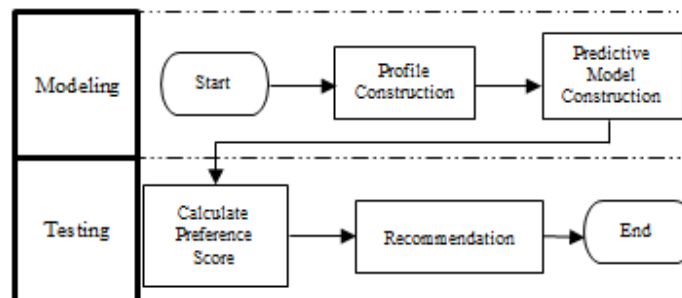


Fig. 2. General scheme of our system.

Data set that used in this system was movie rating data set from Movielens site, namely 100K data set, it has 100.000 rating with total of 943 users and 1682 items. In the experiment, data set were separated into four partitions [3]. The partitions were divided randomly into a half as a new user and the other half as existed user. And for data item, they were divided randomly into a half as new item and the other half as existed item. Fig. 1 illustrates how data were distributed. Part I (including existed data user and existed data item) is used as modeling data and part II (including new user data and existed item), part III (including existed data user and new item), and part IV (new data user and new data item) used as testing data.

1) Modeling

a) Profile Construction

Profile construction produced user profiles, item features, and matrix of rating from user to item. In user profile construction, user vector was created from user data that contains demographic information (gender, age, and occupation) and a history of user habits until a certain time. In item features, an item vector was created from item data set that contains static features, namely film's release year and genre films. In matrix of user-item rating, rating was used to observe relation between user and item.

b) Predictive Model Construction

To get a predictive model of the system, first, calculate the optimal weight value using equation (3) and data in partition I, then use that value to calculate the user's preference score for each item.

2) *Testing*

In testing process, preference score calculated using equation (2), data in partition II, III, and IV, and optimal weight value from modelling process. Then, recommendation for user was given based on the best preference score.

IV. RESULTS AND DISCUSSION

A. *Testing Scenario*

To test the system performance in this study, we used several testing scenarios and the result measured using nDCG. Different from [3], in scenario 1 we conducted experiment to find optimal weight value using several λ values: 0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000, and 100000. λ is a parameter in equation (3) as tradeoff between empirical error and model complexity. The resulted weight value then used to calculate preference score value of each user for each item in training data. To determine the optimal weight value, performed by calculating the nDCGk value, with k=5 and 1. To remove performance bias, nDCG calculated based on sampling which executed 200 times for each user, then it was calculated to find the average value [3]. While in scenario 2 was performed an experiment for testing the system performance based on the correctness of generated item ranking, measured in nDCGk. The test was performed by using the most optimal weight value from scenario 1 and applied on data set in partition II, III and IV. Where k value that is observed was k=1 and k=5. In this scenario also calculated the value of standard deviation from recommendation result to observe the rating distribution from recommendation result.

B. *Result*

Testing scenario 1 was applied on data set partition I with existed user and existed item. In scenario 1 performed calculation using nDCG. The greater nDCG values (range 0 to 1.0) indicates the recommendation result which generated by system is closer to original value. In experiment performed at scenario 1, the value of nDCG that were obtained is shown on Fig. 3. From that result can be seen that λ value and k affected the result of nDCG. Based on the experiment, acquired the best nDCG₁ value = 0.84183 with λ value of 1000 and nDCG₅ = 0.84204 with λ value of 1000. So, the best λ is when λ reach value of 1000. Because when λ value was smaller than that, mean that the result of recommendation depend on rating values, this caused the decrease of system performance. When λ value was greater than that, mean that the recommendation result was less observed the rating value, this also caused the decrease of system performance. The best λ value from scenario 1 then used in scenario 2.

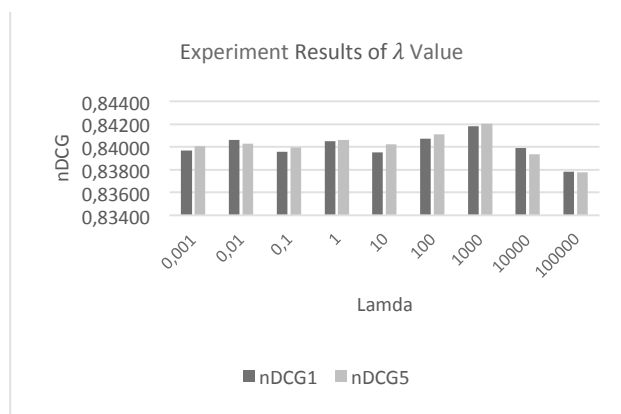


Fig. 3. Experiment result of λ value.

The testing scenario 2 was performed at data partition II, III, and IV using λ value of 1000. The result was shown in Table I. From the result can be seen that the best nDCG score for testing data was 0.848492, when

using data in partition II. It was because the data in partition II used data of existing item, as a result, the recommendation result was better than the recommendation of new item. The value of nDCG for k=1 and k=5 in each partition had the same result, it means, the result of the recommendation for k = 1 had the same performance as for k = 5. The value of nDCG that was obtained for all partition was more than 0.83, it means that this recommendation system was good to solving cold-start problem [3].

TABLE I. NDCG OF TESTING SCENARIO 2

Testing Data	Best λ	nDCG ₁	nDCG ₅
Partition II	1000	0.848404	0.848492
Partition III	1000	0.839270	0.838997
Partition IV	1000	0.837326	0.837241

The result of standard deviation for scenario 2 was explained in Table II. From the result, the average distribution value of rating for partition II, III, and IV respectively were +/- 1.183 with mean = 3.894, +/- 1.249 with mean = 3.794, and +/- 1.229 with mean = 3.784. Based on those standard deviations, it means that data in partition II, III, and IV had a dispersion rating around 3 to 5 (rounded value). Therefore, it can be concluded that the recommendation made by system had a dispersion rating around 3 to 5.

TABLE II. STANDAR DEVIATION OF RATING IN SCENARIO 2

Testing Data	STD	Mean
Partition II	1.183	3.894
Partition III	1.249	3.794
Partition IV	1.229	3.784

V. CONCLUSION

From the experiment conducted in this study obtained that the best λ value for optimal weight was 1000. User profile and item feature that were constructed to build prediction model were able to give a good recommendation by the best value of nDCG was 0.848492. Thus, recommendation for new user from the existing item gives better result than recommendation new item for existing user and recommendation new item for new user. The standard deviation of rating resulted by the recommendation system was 1.24, the average was 3.82. Consequently, the rating distribution of recommendation result is around rating 3 to 5. From the result, it can be concluded that the system gives a good recommendation result to solve cold start problem in movie recommendation system.

For further improvement, it may be considered to add another movie's attribute such as actors or movie studio, to better suit the user preference, or utilize dynamic content such as movie popularity and freshness. Besides that, the use of heuristic algorithm to find the optimal λ value also can be considered to improve the result of recommendation.

REFERENCES

- [1] Ricci, Francesco, Lior Rokach, and Bracha Shapira. *Introduction to recommender systems handbook*. Springer US, 2011.
- [2] Adomavicius, Gediminas, and Alexander Tuzhilin. "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions." *Knowledge and Data Engineering, IEEE Transactions on* 17.6 (2005): 734-749.
- [3] Park, Seung-Taek, and Wei Chu. "Pairwise preference regression for cold-start recommendation." *Proceedings of the third ACM conference on Recommender systems*. ACM, 2009.

- [4] Pazzani, Michael J. "A framework for collaborative, content-based and demographic filtering." *Artificial Intelligence Review* 13.5-6 (1999): 393-408.
- [5] Basilico, Justin, and Thomas Hofmann. "A joint framework for collaborative and content filtering." *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2004.
- [6] Billsus, Daniel, et al. "Adaptive interfaces for ubiquitous web access." *Communications of the ACM* 45.5 (2002): 34-38.
- [7] Järvelin, Kalervo, and Jaana Kekäläinen. "IR evaluation methods for retrieving highly relevant documents." *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2000.

